VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky

Analýza využití virtualizačního nástroje pro SIP a IAX

Analysis of the Virtualization Tool Impact on SIP a IAX

2013 Martin Mikoláš

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky

Zadání bakalářské práce

Student: **Martin Mikoláš**

Studijní program: B2647 Informační a komunikační technologie

Studijní obor: 2612R059 Mobilní technologie

Téma: Analýza využití virtualizačního nástroje pro SIP a IAX
Analysis of the Virtualization Tool Impact on SIP and IAX

Zásady pro vypracování:

Řešitel se seznámí s virtualizačním nástrojem VMware, ústřednou Asterisk a protokoly SIP a IAX. Cílem práce bude vytvořit trunkové spojení prostřednictvím SIP a IAX mezi dvěma ústřednami Asterisk. Dále řešitel bude sledovat kvalitu hovoru při spojení přes SIP a IAX. Na závěr budou dosažené výsledky porovnány a vyhodnoceny.

1. Popis virtualizačního nástroje VMware.
2. Popis ústředny Asterisk.
3. Popis protokolů SIP a IAX.
4. Vytvoření trunkové spojení SIP a IAX mezi ústřednami Asterisk.
5. Porovnání a vyhodnocení kvality hovorů při spojeních přes SIP a IAX.

Seznam doporučené odborné literatury:

1. BASTIAANSEN, Rob. Rob's Guide to Using VMware: Second Edition.
2. MEGGELEN, Jim Van, Jared SMITH a Leif MADSEN. Asterisk: The Future of Telephony.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Karel Tomala**

Datum zadání: 16.11.2012

Datum odevzdání: 07.05.2013

prof. RNDr. Vladimír Vašinek, CSc.
vedoucí katedry

prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 *Studijního a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava.*

V Ostravě 6. května 2013 .....

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 6. května 2013 .....

Abstrakt

Bakalářská práce se zabývá možností propojení pobočkové ústředny Asterisk mezi pevným a virtuálním strojem za použití protokolu SIP a IAX, kterým je věnována pozornost a jsou popsány v jedné z kapitol. Dále jsou popsány způsoby virtualizace pomocí nástrojů společnosti VMware a popis telefonní ústředny Asterisk. V rámci bakalářské práce byly vytvořeny konfigurační soubory, pomocí kterých se ústředna konfiguruje, následně bylo provedeno spojení mezi ústřednami. Závěrem práce je uvedeno vyhodnocení hovorové kvality pro jednotlivé protokoly SIP a IAX.

Klíčová slova: VMware, Asterisk, SIP, IAX, PESQ, MOS

Abstract

Bachelor thesis deals with the possibility of creating connection between two communication servers Asterisk on real computer and virtual computer with usage of protocols SIP and IAX, which are described in one of the chapters. Next point deals with ways of virtualization with help of tools from company VMware and description of communication server Asterisk. Part of this thesis deals with creating of configuration files for configuring Asterisk server, followed by establishing connection between communication servers. At the end of this thesis are listed qualities of each connection for both protocols SIP and IAX.

Keywords: VMware, Asterisk, SIP, IAX, PESQ, MOS

Obsah

Seznam tabulek

Seznam obrázků

1 Úvod

Současná doba vykazuje značný pokrok na poli virtualizace. Existují řešení virtualizace osobních počítačů, serverů a serverových polí. Avšak začíná zasahovat i do pole telekomunikační služby, tedy přenosu hlasu a videa přes internet, která se označuje anglickým názvem Voice Over IP (VoIP). Jedním ze způsobů, jak tuto službu vytvořit a spravovat, je volně dostupná telefonní ústředna Asterisk. Samotná VoIP technologie umožňuje přenos dat pomocí řady protokolů, mezi nimiž se nachází zatím nejrozšířenější protokol SIP (Session Initiation Protocol) a nejnovější přírůstek do rodiny VoIP protokolů, protokol IAX (Inter-Asterisk eXchange), který jak již název napovídá, je speciálně navržen pro Asterisk.

V této práci se budeme zajímat o možnost virtualizace pobočkové telefonní ústředny Asterisk, její využití pro komunikaci pomocí SIP a IAX protokolu. Podíváme se na základní programy pro virtualizaci poskytované firmou VMware. Rozebereme ústřednu Asterisk, její základní vlastnosti a využití. Dále si popíšeme protokoly SIP a IAX, jejich způsob komunikace, výhody a nevýhody.

Ukážeme nastavení telefonních ústředen Asterisk, tak aby byly schopny mezi sebou komunikovat pomocí SIP a IAX protokolu. Uvidíme způsob vytvoření tzv. Dialplanu, který ústředně Asterisk definuje, jakým způsobem má nakládat s příchozími a odchozími hovory. Následuje samotné provedení hovoru a co nás nejvíce zajímá u telefonních hovorů, vyhodnocení kvality hovoru, tedy jak je hovor srozumitelný pro lidský sluch.

2 Teorie

2.1 Popis virtualizačího nástroje VMware

VMware je jedna z největších a nejrozšířenějších společností na poli virtualizace a vývoji virtualizační nástrojů pro architekturu x86/x64. Mezi nejznámnější nástroje můžeme zařadit řešení pro stolní počítače VMware Workstation, VMware Player a řešení pro serverové počítače VMware vSphere. V kapitole čerpám ze zdrojů [1, 2, 3].

Produkty VMware rozdělují stanice na dva typy. Hostitelskou stanici, která provádí výpočty, stará se o správu virtualních vstupů/výstupů a hosta, který emuluje veškeré prostředky, které virtuální systém potřebuje, přesněji procesor, grafický adaptér, paměti, síťový adaptér, ovládače disků. Potřebné prostředky jsou odebíráný z hostitelského počítače, tudíž počet počet běžících virtualním strojů je omezen hardwerovými prostředky hostitelského stroje. Pro všechny hosty je dostupný balíček VMware tools, který zlepšuje výkon.

Díky této virtualizaci má virtuální stroj pokaždé stejný hardware. Pokaždé má stejný síťový adaptér, stejný grafický adaptér a pevné disky. Což umožňuje snadný přenos virtuálních strojů mezi hostitelskými systémy, dokonce lze přenášet virtualní stroje mezi operačními systémy Linux a Windows. Nikdy se nepozmění nastavení virtuálního stroje, který je uložen ve třech typech souborů:

- .vmx soubor

  Tento soubor obsahuje konfiguraci virtuálního stroje. Jsou v něm uloženy informace o použitých pevných discích, nastavení síťového spojení, počet přiřazene RA paměti a počet CPU.

- .vmdk soubor

  Virtualní pevný disk, vytvořený pro virtuální stroj, je ve skutečnosti soubor, do kterého se ukládá vše, co se provede na virtuálním stroji.

- .nvram soubor

  Každý počítač má BIOS a VMware virtuální stroje nejsou výjimkou. Tento soubor má stejnou funkci jako NVRAM, tedy ukládají se do něj BIOS informace o bootovacím pořadí pevných disků, diskety a CD-ROM.

2.1.1 VMware Tools

Balíček přidává nástroje a ovladače, které zlepšují výkon a funkčnost virtuálního systému. Jde nainstalovat na všech podporovaných verzí operačních systémů. Také vylepšuje způsob ovládání přidáním sdílených složek a kopírovaní souborů přetažením z hostitelského počítače do virtuálního. Přidává nástroje pro synchronizaci času s hostitelským

systémem a automatickou detekci kurzoru, která odstraňuje nutnost mačkání tlačítek, když jsme chtěli přepnou mezi virtuálním počítačem a hostitelským.

2.1.2 VMware Player

Podle licence je VMware Player zdarma stažitelný software pro domácí použití, který se nesmí využívat pro výdělečné účely. Dá se o něm uvažovat jako o omezené verzi VMware Workstation. Omezen je počet virtualních strojů, které mohou být současně spuštěny na dva, omezení platí i na počet procesorů a velikost paměti.

Díky těmto omezením je většinou využíván na rychlé odzkoušení nových verzí operačních systémů. které se dokonce nachází na fanouškovských stránkách, které umožňujou stažení hotových, nainstalovaných a nakonfigurovaných operačních systémů. Využití je také při předváděcích akcích. Tento program splňuě očekávání a jeho hlavní cíl, kterým je rozšířit značku VMware mezi lidi, co začínají s virtualizací.

Umožňuje nainstalovaní x86 nebo x64 architektury nehledně na to, jakou architekturu ma hostitelský počítač. Takže mužeme nainstalovat 32b virtualní systém na 64b hostitelký systém a naopak.

V minulosti bylo mezi VMware player a VMware Workstation hodně rozdílů ve funkcích a nástrojích, ale díky konkurenci ze strany dalších poskytovatelů virtualizačních nástrojů byli nuceni tyto rozdíly zredukovat.

Jedna z takových funkcí se jmenuje SnapShot(snímek), můžeme ji přirovnat k funkci Obnovení systému i Windows. SnapShot je však mnohem lepší, protože vrací virtuální systém do přesného stavu, kdy byl SnapShot pořízen. SnapShot lze vytvořit ručně a podle potřeby jej nahrát, samotné nahravání netrvá dlouho, protože se veškeré změný a snímky ukládají na pozadí do speciálního souboru.

Další funkce se nazývá Unity a umožňuje propojení plochy hostitele a virtuálního stroje takovým způsobem, že otevřená okna virtuálního stroje se přenesou na plochu hostitelského systému a okno virtulaního stroje přejde do pozadí. Tím se umožní rychlejší práce na obou systémech.

2.1.3 VMware Workstation

První produkt, který firma VMware vydala v roce 1999 byla ve verzi 1.0, avšak první použitelná verze 3.0 vyšla až v roce 2001. Dnešní verze se jmenuje VMware Workstation 9, jenž vyšla v roce 2012 ve verzi 9.0 a jak je zvykem, přináší lepší vlastnosti a funkce než předchozí verze.

Workstation je zaměřen na stolní počítače, lze jej nainstalovat na operační systémy Windows a Unix, rovněž podporuje jejich virtualizaci, v současné době podporuje přes 700 operačních systémů a toto číslo roste každým dnem.

Workstation stejně jako VMware Player umožňuje spouštět a spravovat několik virtuálních systémů najednou, ale stejně jako VMware Player má taktéž omezení na počet současně běžících systémů a ten je stanoven na deset.

Oproti VMware Player má funkci klonování, která se vyskytuje například u Hyper-V Windows serveru. Umožňuje provést naklonování stávajícího virtuálního stroje do nového s jiným názvem, ale se stejnou konfigurací. Můžeme také provést Linked Clone, který pouze odkazuje na základní virtuální stroj a ukládá provedené změny do vlastního souboru, čímž šetří místo. V praxi se tato funkce může využít například při testování kompatibility softwaru. A to takovým způsobem, že na jednom klonu bude Windows Vista service pack 1 a na druhém Windows Vista service pack 2 a programátor bude hledat chyby v běhu programu.

2.1.4 VMware vSphere 5

VMware vSphere 5 rozděluje četnou řadu nástrojů předchozí generace do několika mnohem podrobnějších, čímž umožňujou VMware správcům ještě vetší kontrolu nad způsobem rozdělování výpočetního výkonů jednotlivým virtuálním strojům. S dynamickou správou prvků, spolehlivostí, systémem odolným proti chybám, zálohovacími nástroji a rozdělenými nástroji pro správu, mají v jednom balíčku IT správci všechny potřebné nástroje pro chod firemní sítě s několika servery po firmy s tisíci servery. Tento balíček obsahuje následující produkty a prvky:

- Produkty:
  - VMware ESXi
  - VMware vCenter Server
  - vSphere Update Manager
  - VMware vSphere Client and vSphere Web Client
  - VMware vShield Zones
  - VMware vCenter Orchestrator

- Prvky:
  - vSphere Virtual Symmetric Multi-Processing
  - vSphere vMotion and Storage vMotion
  - vSphere Distributed Resource Scheduler

- vSphere Storage DRS
- Storage I/O Control and Network I/O Control
- Profile-Driven Storage
- vSphere High Availability
- vSphere Fault Tolerance
- vSphere Storage APIs for Data Protection and VMware Data Recovery

VMware ESXi

Jádrem produktu vSphere je hypervizor, který slouží jako základní virtualizační vrstva, na které jsou postavené další virtualizační nástroje vSphere. V dřívějších verzích vSphere, byl hypervizor dostupný ve dvou formách VMware ESX a VMware ESXi. I když měly obě formy stejné virtualizační jádro, podporovaly stejné virtualizační nástroje, měly rozdílné architektury. V VMware ESX byla použita linuxová konzole, skrz níž mohl správce pracovat s hypervizorem.

Na druhou stranu VMware ESXi je další generací základní virtualizační vrstvy, která nepoužívá linuxovou konzoli. Díky tomu má ESXi velikost pouze 70 MB, ale i přes absenci konzole umožňuje chod stejných virtualizačních nástrojů. To je umožňeno tím, že virtualizační jádro není součástí konzoly, ale samotnou částí zvanou VMkernel, která se stará právě o přístup virtualních strojů k fyzickým prostředkům.

VMware vCenter Server

Poskytuje centralizovanou správu pro všechny ESXi virtuální stroje. Dává správcům možnost spravovat, monitorovat a zabezpečit virtuální infrastrukturu z jednoho kontrolního centra, kde se data ukládají do páteřní databáze.

Kromě nastavení a spravování poskytuje Vcenter Server mnohem pokročilejší nástroje vSphere Motion, vSphere Distributed Resource Scheduler, vSphere High Availability a vSphere Fault Tolerance a další výhody:

- Enhanced vMotion Compatibility, která vylepšuje funkčnost a kompatibilitu serverů spojených do clusteru s CPU od firmy Intel a AMD
- Profily hostů, které napomáhají lepší jednotvárnosti konfiguračním profilům
- Storage I/O Control, který umožňuje quality of service na clusteru, tak aby správci mohli zabezpečit dostatečné I/O prosředky pro důležité aplikace a to i během přesycení sítě.

vCenter Sphere má klíčovou roli v jakékoliv implementaci VMware vSpherea je dotupný ve třech variacích:

- vCenter Sphere Essentials, který je integrovaný do vSphere Essentials kits pro nasazení v kancelářích
- vCenter Server Standard, který poskytuje plnou funkčnost vCenter Sphere, včetně správy, pozorování a automatizace
- vCenter Server Foundation, je stejná jako vCenter Server Standard, ale je limitován na správu tří ESXi hostů a neobsahuje vCenter Orchestrator

vSphere Update Manager

vSphere Update Manager je plug-in pro vCenter Server, který pomáhá užívatelům udržovat jejich ESXi hosta a virtuální stroj na nejnovější verzi. vSphere Update Manager ma následující fuknce:

- Vyhledává systémy, které nejsou na nejnovější verzi
- Uživatelem definovaná pravidla pro identifikaci neaktuálních systémů
- Automatická instalace aktualizací ESXi hostů
- Plná integrace s ostatními produkty vSphere

VMware vSphere Client and vSphere Web Client

vSphere Client je aplikace založená na Windowsu, která umožňuje správu ESXi hostů buď přímo, nebo instancí vCenter Server, která nám narozdíl od přímého připojení na ESXi hosta nabídne plnou škálu správcovských nástrojů. vSphere Client nabízí bohaté uživatelské rozhraní(GUI) pro každodenní činnost a pokročilou správu virtuální infrastruktury.

vSphere Web Client poskytuje dynamické uživatelské rozhraní pro správu virtuální infrastruktury v internetovém prohlížeči, a tím umožňuje správcům vSphere spravovat jejich infrastrukturu, bez nutnosti úplné instalace nástroje vSphare Client. Avšak neobsahuje všechny jeho nástroje.

VMware vShield Zones

vShiled Zones přídává síťovou možnost přidat virtuální firewall, který umožňuje správcům vSphere vidět a spravovat síťovou komunikaci, která probíhá ve virtuální síti. Správci mohou přidávat bezpečnostní práva pro celé skupiny virtuálních strojů, které budou platit i po jejich přemístění mezi hosty.

VMware vCenter Orchestrator

Orchestrarot je automaticky řízené prostředí, které se automaticky instaluje s každou instancí vCenter Server. Správci můžou pomocí toho produktu vytvořit automatické procesy pro služby, které nabízí vCenter Server. Vytvořené procesy mohou mít složitost od jednoduchých po komplexní. VMware také nabízí plug-in pro rozšíření funkčnosti na práci s Microsoft Active Directory, Unified Computing System společnosti Cisco a VMware vCloud Director. To dělá vCenter Orchestrator velice silným nástrojem pro vytváření automatických procesů ve virtualizačním datovém centru.

vSphere Virtual Symmetric Multi-Processing

Prvek vSphere Virtual Symmetric Multi-Processing (vSMP) umožňuje administrátorům virtuální infrastruktury vytvořit virtuální stroje s několika virtuálními procesory. Tedy nevytváří virtuální stroj s několika fyzickými procesory, ale vytváří virtuální procesory uvnitř virtuálního stroje.

To umožňuje instalaci aplikací, které mohou nebo potřebují více procesorů do virtuálního stroje. Organizace využívající vSMP, tudíž mohou virtualizovat ještě více aplikací, bez ztráty výkonu.

vSphere vMotion and Storage vMotion

vSphere vMotion je také známý pod názvem live migration, je to prvek ESXi a vCenter Server, který umožňuje přenášet aktivní virtuální stroj z jednoho fyzického hosta na druhého, bez nutnosti vypnutí virtuálního stroje. Přenos mezi dvěma hosty probíhá bez zpoždění a bez ztráty síťového připojení k virtuálnímu stroji. Tato schopnost přenosu virtuálního stroje mezi fyzickými hosty je v dnešní době silný prvek a má mnoho využití v datacentrech.

Například můžeme uvažovat, že fyzický stroj má nekritický problém na hardwaru a musí být opraven. Správce může jednoduše pomocí vMotion přesunout všechny virtuální stroje na jiný server poté, co byla provedena oprava a server byl zapnut, správce může pomocí VMotion přenést virtuální stroje na původní server.

vSphere Distributed Resource Scheduler

Předchozí vMotion se musí ovládat ručně, to znamená, že správce musí zahájit proces přenosu, ale tuhle činnost umí vSphere Distributed Resource Scheduler (DRS) provádět automaticky a tím vyrovnývat rozložení výpočetní síly po ESXi clusteru.

O ESXi clusteru můžeme říct, že není nic víc, než soubor výpočetního výkonu a paměti všech hostů v clusteru. Poté co se do clusteru přiřadí dva nebo více hostů, začnou společně pracovat na poskytování výpočetní síly a paměti pro virtuální stroje přiřazené do clusteru.

vSphere High Availability

Nejčastější argument proti virtualizaci je: "Co se stane, když server bude mít kritické selhání". To v minulosti narušilo chod jedné aplikace, ale v clusteru to naruší chod mnoha procesů.

Toto se snaží vSphere High Availability (vSphere HA) napravit. Tento element vSphere poskytuje automatický proces pro restartování všech virtuálních strojů, které byly spuštěné na ESXi hostu v době selhání serveru. Aplikace sice budou vypnuté během procesu přenesení a restartu, ale celkový čas bude menší (přibližně tři minuty), než čas potřebný pro restart ESXi hosta.

vSphere HA nepoužívá vMotion pro přenos virtuálních strojů mezi ESXi hosty na rozdíl od DRS, která se používá pro plánovaný přenos. Selhání serveru je neočekávané a tím pádem nemůžeme provést nastavení DRS (vSphere Distributed Resource Scheduler).

2.2 Popis úsředny Asterisk

2.2.1 Co je asterisk?

Projekt Asterisk začal v roce 1999, kdy jeho tvůrce Mark Spencer vydal základní kód pod GPL (Gerenal Public Licence) open source licencí a od té doby byl vylepšen a otestován tisíci uživateli. Dnes je Asterisk spravován společným úsilím společností Digium a jeho komunitou. V této kapitole je čerpáno ze zdrojů [4, 5, 13, 14, 15].

Asterisk je zdarma poskytované open source řešení aplikace pro komunikaci. Pomocí Asterisku můžeme změnit normální stolní počítač na komunikační server. Asterisk rovněž nabízí velkou spoustu služeb a právě díky tomu, je hojně využíván v komerční oblasti od malých podniků po zákaznické centra a vládní orgány. Nejčastější využití je v Business Phone Systems IP PBXs (pobočková ústředna IP PBXs), ale má i mnoho dalších, například:

- VoIP gateway
- Business Phone System (IP PBX)
- Voicemail Server
- Conference Bridge
- Call Center
- IVR Server

VoIP gateway

VoIP gateway (VoIP brána) se používá pro spojení dvou rozdílných systémům jako starší telefonní systémy (BPX) a VoIP. Spojení se navazuje přes digitáln nebo analogové trunk porty. Volání z BPX do vnějšího světa se konvertují na VoIP volání a posílá se přes internet k VoIP poskytovateli. Příchozí VoIP volání se konvertují na příslušný protokol a pošlou do BPX.

Brány se rozdělují na 2 typy. Analogová brána, která konvertuje mezi VoIP protokoly, a tradiční analogovou telefoní linkou a digitální. Digitální brána konvertuje VoIP na digitální telefoní služby: T1, E1, PRI a BRI.

Business Phone System (IP PBX)

O BPX můžeme říct, že je centrálním bodem pro přepínání hovorů v pobočce. Kontroluje komunikaci po vnitřní síti a vystupuje jako brána do vnější sítě. Každá ústředna je tvořena dvěma částmi, stanice a propojení. Propojení, neboli trunk, jsou spoje s globální veřejnou telefonní sítí (public switched telephony network (PSTN)). Stanice nejsou nic jiného než telefony nebo jiné koncové zařízení (fax, modem).

Voicemail Server

Tato služba umožňuje zanechat hlasovou zprávu volanému a probíhá následovně. Proces nahrávání správy se spustí, když nejsme schopni se dovolat volanému. Aplikace pro sběr tohoto typu správ dostane data, která udávají, kdo byl volaný následně přehraje uvítání a volající může nahrát zprávu. Voicemail služba odešle příkaz BPX, aby zapnul indikátor čekající zprávy na telefonu volaného. Poté co volaný dostane tuto informaci, tak si může vyzvednou zprávu. Starší systémy vyžadovaly, aby volaný zavolal na záznamníkovou službu, přihlásil se pomocí předvolby a čísla a pak si poslechl vzkaz. Novější systému umožňují, aby si volaný vyzvedl vzkaz na svém počítači nebo mobilu pomocí "visual voicemail".

Conference Bridge

Conference Bridge umožňuje vytvořit hovor mezi skupinou lidí, kteří se připojují do virtuální místnosti. Ve virtuální místnosti můžou být desítky až stovky účastníků najednou. Konferenční systém podporuje chod více virtuálních místnosti najednou a limitací je hardware. Jednotlivé místnosti lze zabezpečit PIN kódem.

Call Center

Telefonická centra jsou speciální kanceláře postavené tak, aby dokázaly obsloužit velké množství telefonních hovorů typu zákaznické podpory, zákaznického centra, telemarketing a dotazníkové služby.

Telefonií centra využívají speciální vybavení pro dosažení maximální produktivity, například automatická distribuce hovorů (ACD), nahrávání hovorů, automatické vytáčení, možnost nahrát stížnost zákazníka.

Asterisk je silný nástroj pro tvorbu telefonních center. Spolu s podporou řazení hovorů, IVR, vzdálené vytáčení, nahrávání hovorů, živý dozor nad hovory a jejich hlášení. Asterisk tedy obsahuje všechno pro tvorbu telefonního centra, kde malé telefonní centra jsou tvořeny jedním Asterisk serverem a velké firmy mají klustery těchto serverů.

IVR Server

Interactive Voice Response (IVR) je technologie, která poskytuje automatickou interakci s volajícím. Volající má možnost pracovat se systémem použitím svého hlasu nebo číselníku. Jednoduchým příkladem je hlasové menu, kde volající odpovídá zadáváním čísel, čímž se odstraní potřeba živého operátora.

Asterisk obsahuje spoustu funkcí, které z něj dělají silného IVR poskytovatele. Patří mezi ně nahrávání a přehrávání audia, databázový a webový přístup, spojení s kalendářem, rozpoznávání hlasu a jeho syntéza. Právě tyto funkce a fakt, že Asterisk je zdarma stažitelný program, který nemá žádné licenční poplatky a může pracovat na levném hardwaru, jej odlišují od ostatních placených IVR platforem.

2.2.2 Podporované technologie

Asterisk je vytvořen tak, aby bylo přidávání nových rozhraní, technologií co nejjednodušší, můžeme říct, že jeho životním cílem je podporovat všechny současné technologie a technologie, které budou teprve vytvořeny. Podporované rozhraní se rozdělují na tři skupiny:

- Zaptel Pseudo TDM interfaces
- Non-Zaptel hardware interfaces
- Packet voice protocols

Packet voice protocols

Sada standartních protokolů pro komunikaci přes paketové (IP a Frame Relay) sítě, jsou to jediná rozhraní, která nepotřebují specializovaný hardware. Jsou to:

- Session Initiation Protocol (SIP)
- Inter-Asterisk eXchange (IAX) verze 1 a 2
- Media Gateway Control Protocol (MGCP)
- ITU H.323
- Voice over Frame Relay (VOFR)

2.2.3 Podporované kodeky

Asterisk podporuje tyto kodeky:

- ADPCM
- G.711 (a-Law u-Law)
- G.722
- G.726
- G.729a
- GSM

REGISTER	Registrace účastníka
INVITE	Sestavení relace
ACK	Potvrzení INVITE a zahájení relace
CANCEL	Přerušení nesestavené relace
BYE	Ukončení relace
OPTIONS	Požadavek na možnosti volajícího

Tabulka 1: Metody SIP protokolu

1xx	Průběh
2xx	Úspěch
3xx	Přesměrování
4xx	Chyba klienta
5xx	Chyba serveru
6xx	Fatální chyba

Tabulka 2: Kategorie SIP zpráv

2.3 Popis protokolů SIP a IAX

2.3.1 Protokol SIP

SIP je textový protokol založený na HTTP (HyperText Transport Protocol) a SMTP (Simple Mail Transport Protocol) protokolu. SIP byl vytvořen IETF Multiparty Multimedia Session Control jako část Internet Multimedia Conferencing Architecture. SIP je využívaný pro peer-to-peer komunikaci, tedy komunikaci, ve které jsou si oba volající rovni. V kapitole je čerpáno ze zdrojů [6, 7, 8, 13, 14, 15].

I přes fakt, že SIP je peer-to-peer komunikace, používá client-server model podobný HTTP. SIP klient vytvoří SIP žádost, na kterou SIP server odpoví vygenerováním odpovědi. RFC 3261 definuje základních šest SIP žádostí, která jsou uvedeny v tabulce 1.

Odpovědi SIP jsou v číselné formě a jsou rozděleny do 6 kategorií. Každá kategorie se rozpoznává podle první číslice, přehled se nachází v tabulce 2.

2.3.1.1 Síťové prvky SIP

SIP má tři základní síťové prvky user agents, servery a location services.

User Agents

User agents jsou koncovými zařízeními v SIP síti. Vytvářejí SIP žádosti pro sestavení spojení a posílají/příjmají data. Mohou to být SIP telefony nebo SIP klient na počítači.

User agent se skládá ze dvou částí, user agent client (UAC) a user agent server (UAS). UAC je část, která vytváří žádosti a na tyto žádosti vytváří UAS odpovědi.

Servery

Servery jsou zprostředkovatelské zařízení v SIP sítích, které pomáhají user agents při sestavování relace. RFC 3261 definuje tři typy serverů SIP proxy, redirect a registrar server. Tyto servery poskytují pouze signalizaci a nemohou vytvářet SIP žádosti.

- SIP Proxy - Zpracovává žádosti user agents, jiného proxy, které následně přeposílá dál do sítě.
- Redirect Server - Zpracovává žádosti user agents a proxy. Vrací odpověď o přesměrování s kódem 3xx, který udává, kde by měl být požadavek opakován.
- Registrar Server - Zpracovává SIP žádosti o registraci a upravuje informace o user agent v location service, nebo jiné databázi.

Location Services

Lokační služba je podle RFC 2543 databáze, která obsahuje informace o uživatelích, jejich URI, IP adresu, skripty, možnosti a jejich nastavení. Také může obsahovat informace o směrování v SIP síti, lokace proxy a výstupních bran. User agents nemanipulují přímo s lokační službou, ale využívají prostředníku proxy, redirect, nebo registrar server.

2.3.1.2 Struktura protokolu

SIP je strukturován jako protokol se čtyřmi vrstvami, které popisují jednotlivé procesy při přenosu zpráv, avšak toto rozdělení je pouze pro účely popisu funkčnosti jednotlivých částí spojení a nemá žádný vliv na samotnou implementaci protokolu.

Nejnižší vrstvou SIP protokolu je syntaxe a dekódování. Samotné dekódování používá gramatiku Backus-Naur Form.

Druhá vrstva je transportní. Definuje jak klient posílá žádosti a jak přijímá odpovědi, také jak server přijímá žádosti a jak odesílá odpovědi po síti. Všechny SIP prvky obsahují přenosovou vrstvu.

Třetí vrstva se nazývá transakční a je základní komponentou SIP protokolu. Vrstva má uživatelskou část značenou „lientská transakce"a serverovou část „serverová transakce". Obě tyto části jsou vyjádřeny konečným stavovým automatem, který je speciálně navržen pro každou žádost. Jejím účelem je provádět žádosti klientů odeslané na serverovou transakci pomocí transportní vrstvy a poté odeslat všechny odpovědi serverové transkace zpátky ke klientovi. Prováděcí vrstva se stará o přeposílaní dat aplikační vrstvy, přiřazování odpovědí žádostem a vypršení doby aplikační vrstvy. Jakékoliv provedení žádosti user agent client (UAC) probíhá řadou transakcí. Transakční vrstvu obsahují pouze user agents (UA) a stavové proxy.

Poslední vrstvou je vrstva transaction user (TU). O každém SIP prvku, kromě bezestavové proxy, můžeme uvažovat jako o transaction user. Při tvorbě TU žádosti se vytvoří instance klientské transakce, která se následně předá do žádosti spolu s cílovou IP adresou, portem a kanálem, do kterého se samotná žádost posílá. Tato žádost pak může být zrušena žádostí CANCEL, která serveru ohlásí, aby tuto žádost nadále nevykonával, navrátil svůj stav do bodu před příchodem žádosti a vytvořil chybové hlášení, které odpovídá typu žádosti.

Jednotlivé SIP prvky (user agent client a server, stavové a bezestavové proxy a registar servery) obsahují jádro, které je odlišuje od sebe navzájem. Všechny jádra krom bezestavové proxy jsou transaction user (TU). Chovaní UAC a UAS jádra záleží na metodě, ale mají některé společné pravidla pro metody. Pravidla pro UAC stanovují vytváření žádostí, na druhou stranu UAS pravidla určují způsob zpracování žádosti a vytvoření odpovědi.

Nejdůležitější metodou SIP protokolu je metoda INVITE, která je použita pro sestavení spojení mezi účastníky. Samotné spojení je souborem účastníků a datovým tokem mezi nimi pro účely komunikace.

2.3.1.3 SIP komunikace

Samotná komunikace začíná, když user agent vytvoří žádost INVITE, která žádá server o vytvoření spojení. Zpráva dorazí na UAS (user agent server), ten se rozhodne, jestli příjme žádost a odpoví třídou zpráv 2xx, nebo odmítne přijmout žádost a odpoví zprávou 3xx, 4xx, 5xx, nebo 6xx, třída těchto zpráv závisí na důvodu odmítnutí.

2.3.2 Protokol AIX

Protokol Inter-Asterisk eXchange umožňuje VoIP komunikaci mezi servery a mezi servery a klienty, kteří používají AIX protokol. IAX byl vytvořen jako open source řešení společností Digium pro komunikaci mezi Asterisk servery, avšak není omezen pouze na Asterisk. Je volně dostupný a podporovaný v řadě dalších open source řešení telefoních ústředen. Čerpáno ze zdrojů [9, 10].

2.3.2.1 Výhody protokolu IAX

IAX je robustní a zároveň jednoduchý protokol, který dokáže přenášet téměř všechny druhy přenosů. Avšak je optimalizován pro potřeby VoIP hovorů, kde jsou prioritou nízké nároky na přenosovou linku a malé přeslechy. Právě tyto priority dělají IAX skvělým protokolem pro VoIP, než ostatní protokoly, které mají podrobnější vlastnosti, ale zasahujou do oblastí, které nejsou potřebné v dnešních přenosech dat nebo peer to peer komunikaci. Navíc IAX je vytvořený jako lehký a VoIP přátelský protokol, který využívá menší část přenosového pásma než ostatní protokoly.

IAX je binární protokol speciálně vytvořený aby redukoval přeslechy při přenosu hlasu. V určitých případech se výkon přenosu po lince zhoršuje ve prospěch výkonu přenosu pro jednotlivé hovory. Například, když přenášíme hovor komprimovaný na 8kbit/s a 20 ms packetization, každý datový packet má 20 bytů. IAX zvýší celkovou hodnotu o 4 byty přeslechu a RTP přidá dalších 12 bytů do každé packetu. Binární protokoly jsou také odolné proti buffer-overrun útokům.

Navíc k výkonnosti IAX využívá stejně jako SIP jeden statický UDP port, což ulehčuje práci siťovému administrátorovi, který dokáže snadněji spravovat, prioritizovat a povolovat komunikaci na firewallech. Základní myšlenkou IAX je multiplexování signalové zprávy a komunikaci do jednoho UDP kanálu mezi dvěma počítači. Také používá stejný UDP port pro přenos signálových zpráv a komunikace a díky tomu, že se všechna komunikace hovoru se provádí na stejné point-to-point cestě je nastavení NAT mnohem lehčí než u ostatních protokolů.

2.3.2.2 Nevýhody protokolu IAX

IAX je vysoce efektivní řešení pro mnohé dnešní komunikační potřeby, ale má několik nedostatků. Například IAX používá point-to-point způsob komunikace, který vyžaduje, aby všechny IAX prvky sítě, které jsou potřebné pro sestavení spojení, umožňovaly využití naprosto stejného kodeku. Navíc, kodek se definuje 32-bitovou maskou, která je definovaná v protokolu, takže počet současně využitelných kodeku je limitovaný.

Dalším problémem může být využití jednoho známého UDP portu, dělá z IAX protokolu lehký cíl pro denial-of-service útok. Systém jako VoIP je vysoce citlivý na tento druh útoku.

Centralizovanost je dalším problémem, protože AIX síť se obvykle vytváří způsobem, kde signálové zprávy a komunikace putujou do jednoho centralizovaného serveru. Což umožňuje zlehčenou správu komunikace po síti, ale omezuje rozšiřitelnost sítě. Tento problém se vyskytoval v dřívějších verzích IAX, ale byl vyřešen možností rozdělit kanály pro přenos signálových zpráv a samotnou komunikaci.

2.3.2.3 Popis protokolu IAX

IAX je peer-to-peer, VoIP protokol. Může registrovat lokace, vytvářet, spravovat, ukončovat multimediální komunikaci a přenášet samotnou komunikaci po síti. Protokol je navržen a optimalizován pro přenos komunikace pomocí Internetových protokolů. Nejnovější verze je IAX2.

Jak bylo zmíněno v kapitole o výhodách IAX protokolu, IAX využívá multiplexování signálových zpráv a komunikace do jednoho UDP portu pro komunikaci mezi dvěma hosty. Tohoto je dosáhnuto použitím stejného známého UDP portu 4569 pro všechny typy IAX komunikace. To zaručuje průhlednost NAT, což je výhodou oproti jiným přenosovým protokolům jako je SIP.

Zakladní jednotkou komunikace v IAX je „Frame". Existuje několik tříd Framů, ale v základě se dají rozdělit na „Full Frames", které přenáší data signálových zpráv a kontrolní data. Další je „Mini Frames", které přenáší samostatnou komunikaci, „Meta Frames"jsou používaný pro volání trunku, nebo video komunikaci.

IAX podporuje využití bezpečnostních prvků povolením několika způsobů ověřování a přihlašování uživatelů, také povoluje několik metod pro peer registraci. IAX také definuje zakladní rozhraní pro šifrování.

IAX je optimalizovaný peer-to-peer protokol. Když je vytvořeno spojení mezi dvěma IAX klienty a když se přenosový peer rozhodne, že již nemusí zůstat v cestě volání, může zahájit změnu cesty volání, čímž odstraní sám sebe z vytvořené cesty volání, avšak proces změny není hotov dokud se nepotvrdí, že oba volající mohou bez problému komunikovat.

2.3.2.4 IAX komunikace

Sestavení IAX komunikace pro hovor mezi dvěma peery začíná vysláním zprávy NEW s číslem nebo jménem vzdáleného peera. Vzdálený peer může odpovědět credentials challenge (AUTHREQ), zprávou odmítnutí REJECT, nebo potvrzením ACCEPT. AUTHREQ zpráva zasílá autentikační schémata a měla by vyústit v odpověď s požadovanými daty zabezpečení v podobě zprávy AUTHREP. Odpověď REJECT značí, že hovor nemůže být vytvořen. ACCEPT značí úspěšné vytvoření hovoru mezi dvěma peery a že může začít komunikace na vyšší úrovni.

- zpráva NEW - Vysílána pro vytvoření spojení. Je unikátní tím, že v hlavičce nepotřebuje destination call identifier, který se vytváří až po uspěšném navázaní spojení, které je identifikované přijetím odpovědi ACCEPT.
- zpráva ACCEPT - Po přijetí zprávy ACCEPT se vysílá se zpráva ACK a volající peer čeká na odpověď ANSWER, HANGUP, PROCEEDING, RINGING, BUSY.
- zpráva REJECT - Zpráva indikuje selhání zpráv NEW, AUTHREP, DIAL nebo ACCEPT. To může být způsobeno špatným uživatelským jménem, selháním autentizace nebo špatným heslem. Po přijetí zprávy REJECT se vyšle ACK na volaného peera a hovor se zruší.
- zpráva HANGUP - Zprávu odesílá jeden z peerů a značí ukončení spojení. Po přijetí HANGUP zprávy IAX peer musí okamžitě odeslat ACK a ukončit spojení, další odesílané zprávy na ukončeném spojení vrací zprávu INVAL, která značí nesprávnost zprávy, protože samotné spojení bylo již ukončeno.
- AUTHREQ - Zpráva se odesílá jako odpověď na zprávu NEW, když je pro sestavení spojení potřeba autentizace jako heslo, uživatelské jméno nebo způsob šifrování. Peer odpovídá zprávou AUTHREP, nebo HANGUP.
- AUTHREP - Zpráva je odpovědí na požadavek AUTHREQ, zasílá požadované informace o zabezpečení a vyžaduje odpověď ACCEPT, kdy jsou požadované informace správné, nebo odpověď REJECT, která značí zaslání špatných informací či nedostupnost peera.

2.3.2.5 Přehled typů Frame

Typ Frame IAX je používán pro správu IAX koncových zařízení. Jsou používáný pro IAX signalizaci jako navázání spojení, údržba spojení a zrušení spojení. Mohou také přenášet data komunikace a nepřenášejí data session-specific protocol, to je úkolem Control Frames. Přehled základních IAX framů je v tabulce 3, každý frame má své specifické označení pomocí hex hodnoty.

Hex	Název	Popis
0x01	NEW	Vytvoření nového hovoru
0x02	PING	Ping žádost
0x04	ACK	Potvrzení přijetí
0x05	HANGUP	Přerušení hovoru
0x06	REJECT	Odmítnutí hovoru
0x07	ACCEPT	Přijetí hovoru
0x08	AUTHREQ	Žádost o autentikaci
0x09	AUTHREP	Odpověď na AUTHREQ
0x0a	INVAL	Invalidní zpráva

Tabulka 3: Typy IAX framů

Hex	Název	Popis
0x01	Hangup	Hovor byl ukončen na vzdáleném konci
0x03	Ringing	Hovor čeká na přijetí na vzdáleném konci
0x04	Answer	Hovor byl přijat na vzdáleném konci
0x05	Busy	Vzdálený konec je zaneprázdněn a nemuže přijmout hovor

Tabulka 4: Typy Control framů

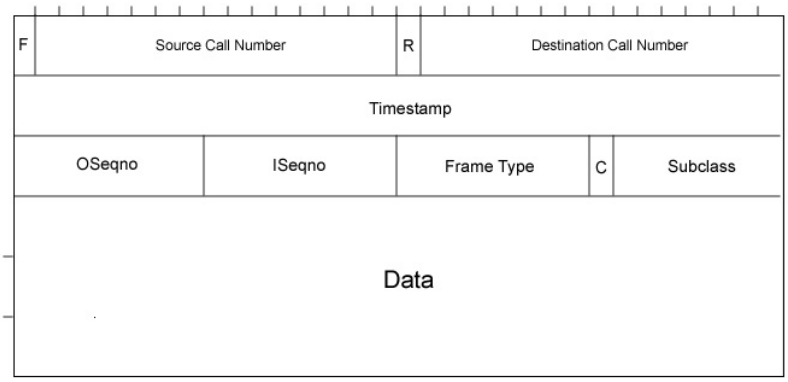
Typ Frame Control přenášejí data o samotném spojení, například stav zařízení. Stejně jako u IAX framů je každý frame označen specifickým číslem v hexadecimální soustavě, jejich základní typy jsou v tabulce 4. Podoba Full Frame rámce lze vidět na obrázku 1.

2.3.2.6 Přenos zpráv

IAX používá pro přenos UDP protokol a používá protokoly aplikační vrstvy pro spolehlivý přenos. Při přenosu se využívají dva typy zpráv, spolehlivá (Full Frames) a nepotvrzená (Mini, nebo Meta framy). Všechny zprávy kromě některých hlasových nebo video zpráv jsou spolehlivé.

Spolehlivé zprávy jsou přenášeny jedním způsobem, kde se zaznamenává číslo zprávy a čas odeslání pro všechny peery účastnící se hovoru. Každý peer sleduje čas pro všechny spolehlivé zprávy a musí opakovat jejích odesílaní, dokuď nepřijme potvrzení přijetí, nebo nevyprší maximální počet opakování.

Při zahajení hovoru je číslo pořadí příchozích a odchozích zpráv nastaveno na nulu. Každá odeslaná spolehlivá zpráva navýší toto číslo o jednu, krom zpráv ACK, INVAL, které nemají vliv na navýšení. Každá odeslaná zpráva obsahuje počet odeslaných zpráv, počet přijatých zpráv a čas, který udává délku hovoru v milisekundách.



Obrázek 1: Ukázka podoby rámce Full Frame [17]

Po přijetí zprávy se kontroluje čas, aby se zajistilo správné pořadí. Když je příchozí zpráva mimo své pořadí, tak se zahodí a pošle se zpráva VNAK pro synchronizaci peeru. U spolehlivých zpráv se kontroluje číslo pořadí a potvrzuje se přijetí zprávy.

Pokuď se zpráva nepotvrdí po určitém počtu pokusů, hovor se považuje za nepoužitelný a musí se zrušit.

3 Praktická část

Teoretický rozbor

Cílem mé bakalářské práce je propojit dvě telefonní ústředny Asterisk pomocí SIP a IAX protokolu. Potřebujeme dva nainstalované operační systémy Linux, na každém počítači nainstalovat ústřednu Asterisk, dále ji nakonfigurovat, provést spojení a přenést testovací zvukové stopy, které porovnám metodou PESQ (Perceptual Evaluation of Speech Quality) a zjistím hodnotu MOS (Mean Opinion Score). Schéma můžeme vidět na obrázku 2.

Prvním krokem je připojení na samotný operační systém, na kterém Asterisk nainstalujeme. V mém řešení máme jeden pevný počítač a jeden virtuální, na který se dostaneme pouze pomocí SSH klienta. Operační systém Linux nám poskytuje tohoto klienta přímo zakomponovaného do terminálu, ale existuje i varianta pro OS Windows, aplikace Putty. Po spuštění aplikace Putty zadáme IP adresu, v našem případě 158.196.244.159, zvolíme protokol SSH a klikneme na Open. V linuxu se pracuje přímo přes terminál příkazem **ssh jmeno_uzivatele@ip_adresa**. V našem případě má tento příkaz podobu **ssh student@158.196.244.159**. Obě varianty ssh nám po úspěšném navázání spojení vypíšou žádost o zadání přihlašovacích údajů. Pokud jsou údaje v pořádku, objeví se shell vzdáleného systému. Práce v něm probíhá v textovém režimu, ve kterém jsou dostupné všechny příkazy vzdáleného systému.

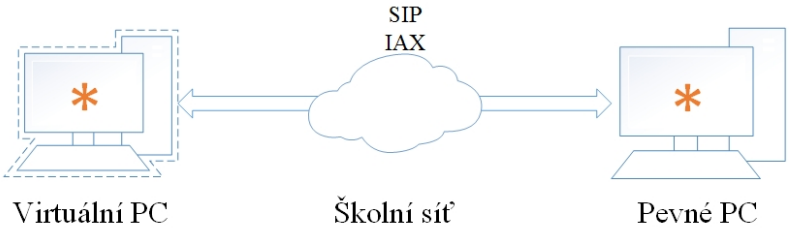
Při samotné instalaci Asterisk musíme dodržet pořadí balíčků. Nejprve se instaluje DAHDI, poté libpri a nakonec Asterisk. Dále při instalaci Asteriku použijeme příkaz **make samples**, který nám vytvoří ukázkové soubory a nastaví Asterisk do podoby, ze které vychází následující postup konfigurace.

3.1 Nastavení telefonní ústředny Asterisk

Pro úspěšné spojení dvou ústředen přes SIP a IAX protokol musíme na každé ústředně nastavit v konfiguračních souborech sip.conf, iax.conf jednotlivé uživatele zvané peery.

Samotné nastavení uživatelů nestačí, pro úspěšnou komunikaci také musíme nastavit, jak se bude nakládat se samotnými příchozími a odchozími hovory na ústřednách. Tato konfigurace se provádí v souboru extensions.conf.

Tyto konfigurační soubory musíme vytvořit pro každý Asterisk. Pro práci s Asteriskem je vhodné pracovat se dvěma okny, v jednom okně se upravují konfigurační soubory a ve druhém okně se připojíme na Asterisk CLI zde provádíme restartovaní upravených modulů, sledovaní funkčnosti, vypisovaní zpráv o stavu Asterisku, jako chyby v konfiguračních souborech, informace o sestavení spojení a stavu jednotlivých uživatelů a kanálů.



Obrázek 2: Schéma praktické části

3.1.1 Konfigurace sip.conf

Pro konfiguraci SIP uživatelů musíme upravit konfigurační soubor sip.conf v lokaci **/etc/asterisk** pro obě ústředny. Zde doporučuji vymazat celkový obsah souboru, abychom omezili vznik chyb při konfiguraci.

Začneme konfigurací pevného počítače. Otevřeme soubor sip.conf pomocí oblíbeného textového editoru a vepíšeme požadovanou konfiguraci. Konfigurace SIP protokolu na pevném počítači vypadá následovně:

```
[ virtual ]
type=friend
host=158.196.244.159
secret=welcome
context=incoming
qualify=3000
```

Výpis 1: Konfigurace SIP protokolu na pevném PC

Položka [virtual] je definice SIP uživatele, se kterým Asterisk na pevném PC bude komunikovat. Definujeme typ uživatele v položce type, zde máme na výběr:

- peer - Uživatel může pouze přijímat hovory Asterisku.
- user - Uživatel může pouze vytvářet hovory skrz Asterisk.
- friend - Definuje uživatele, který může vytvářet i příjmat hovory Asterisku. Tento typ je nejvíce používán a je použit i v našem postupu.

Následuje definice host, kde definujeme pod jakou IP adresou se náš Asterisk na virtuálním stroji připojuje. Secret značí heslo, o které se žádá při sestavení spojení zprávou SIP INVITE. Řádek context definuje seznam akcí, které se nachází v konfiguračním souboru extensions.conf. Nakonec máme řádek qualify, zde máme na výběr možnosti yes, která Asterisku říká, že má provést kontrolu spojení každých šedesát vteřin. Možnost no, která zakazuje kontrolu, a možnost vložení vlastní hodnoty v milisekundách. Použijeme hodnotu 3000 a soubor uložíme, přejdeme do Asterisk CLI a použijeme příkaz **sip reload**, kterým řekneme Asterisku, aby znovu načetl konfigurační soubor.

Name/username	Host	Dyn Nat ACL	Port	Status
virtual	158.196.244.159	N	5060	OK (5 ms)

Tabulka 5: Ukázka příkazu sip show peers

Pro konfiguraci sip.conf na virtuálním stroji použijeme postup popsaný výše, jediná změna je v IP adrese uživatele, kterou přepíšeme na IP adresu pevného stroje.

Po znovunačtení konfiguračních souborů na obou strojích se můžeme v Asterisk CLI přesvědčit o funkčnosti spojení příkazem **sip show peers**, který nám vypíše podrobnosti o vytvořených uživatelích, například v podobě, jakou vidíte v tabulce 5, která zobrazuje výpis tohoto příkazu na pevném počítači.

3.1.2 Konfigurace iax.conf

Postup konfigurace IAX se příliš neodlišuje od postupu konfiguarce SIP. Lokace souborů se opět nachází v **/etc/asterisk** a stejně jako u SIP musíme nastavit obě ústředny.

Začneme konfigurací pevného počítače. Otevřeme soubor iax.conf a vepíšeme naši konfiguraci, která má následující podobu:

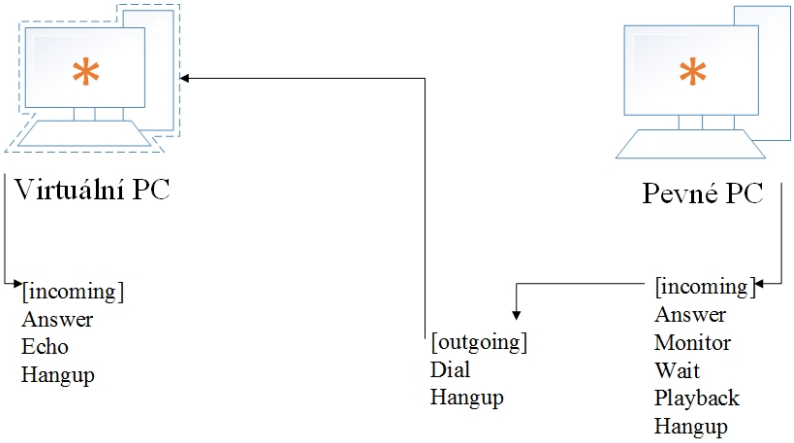
```
[general]
 autokill =yes

[ virtual ]
type=friend
host=158.196.244.159
trunk=yes
secret=welcome
context=incoming
deny=0.0.0.0/0.0.0.0
permit=158.196.244.159/255.255.255.0
 qualify =3000
```

Výpis 2: Konfigurace IAX protokolu na pevném PC

Většina příkazů má stejný význam jako u sip.conf, ale iax využívá pro komunikaci spojení typu trunk, takže musíme do kolonky [general] připsat řádek autokill a nastavit jej na stav yes. To nám zajistí, že data co vysíláme, jsou potvrzeny zprávou ACK, abychom neposílali data do trunku naprázdno. Samozřejmě musíme zapnout trunkové spojení nastavením na hodnotu yes, dále zakázat spojení se všemi nepovolenými ip adesami příkazem deny a povolit příchozí komunikaci z virtuálního stroje příkazem permit. Po napsání příkazů soubor uložíme a přejdeme do Asterisk CLI, kde provedeme znovu načtení konfiguračního souboru iax příkazem **iax2 reload**.

Name/username	Host	Mask	Port	Status
virtual	158.196.244.159 (S)	255.255.255.255	4569 (T)	OK (5 ms)

Tabulka 6: Ukázka příkazu iax2 show peers



Obrázek 3: Přehled konfigurace dialplan

Na virtuálním stroji je konfigurační soubor opět stejný jako u pevného počítače, ale musíme změnit IP adresu hosta a povolenou IP adresu v příkazu permit na IP adresu pevného počítače.

Po znovu načtení konfiguračních souborů na obou strojích se můžeme v Asterisk CLI přesvědčit o funkčnosti spojení příkazem **iax2 show peers**, který nám vypíše podrobnosti o vytvořených uživatelích v podobě, která je zobrazena v tabulce 6. Oproti výpisu SIP vidíme rozdíly, jako (T) u portu, což značí, že máme vytvořené trunkové spojení.

3.1.3 Konfigurace extensions.conf

Modul, který v Asterisku pracuje s tímto souborem se nazývá dialplan. Je vhodné si před samotným psaním konfigurace vytvořit tento dialplan, kde si rozmyslíme, jaké akce budeme používat. Náš diaplan, který budeme vytvářet, je zobrazen na obrázku 3.

Konfigurace extension.conf se odlišuje od konfigurace jednotlivých uživatelů, protože nebudeme definovat vlastnosti, ale přímo způsob nakládaní s hovory, tudíž budeme vytvářet seznam akcí. Konfigurační soubor je opět umístěný v **/etc/asterisk** a opět doporučuji vymazat obsah tohoto souboru.

Začneme konfigurací pevného počítače, otevřeme soubor extensions.conf v oblíbeném textovém editoru a vepíšeme seznam akcí v následující podobě:

```
[globals]

[general]

[incoming]
exten => _X,1,Answer()
exten => _X,2,Monitor(wav,${CALLFILENAME})
exten => _X,3,Wait(5)
exten => _X,4,Playback(XX)
exten => _X,5,Hangup()

[outgoing]
exten => _X,1,Dial(SIP/virtual/${EXTEN},30)
;exten => _X,1,Dial(IAX2/virtual/${EXTEN},30)
exten => _X,2,Hangup()
```

Výpis 3: Konfigurace extensions.conf na pevném PC

Na pevném počítači definujeme dva seznamy akcí incoming a outgoing, kterým se říká context.

Context incoming má na starost monitorovaní hovoru a přehrávání testovací zvukové stopy. Příkaz Answer příjme hovor, příkaz Monitor nahrává zvuky, které se nacházejí na lince a ukládá je ve formátu .wav v lokaci **/var/spool/asterisk/monitor**. Příkaz Wait má nastavenou dobu čekání na pět sekund, ve které se vytvoří samotný hovor, Playback přehraje zvukovou stopu ze souboru, který určíme. Soubor se musí nacházet v lokaci **/var/lib/asterisk/sounds/en** a jméno tohoto souboru se vepisuje do závorky místo XX. Jako poslední je příkaz Hangup, který ukončí spojení.

Context outgoing se stará o samotné vytvoření hovorů. Máme zde příkaz Dial, který se zadává v podobě Protokol/uživatel/klapka, v našem dialplan máme uvedeny příkazy Dial pro oba protokoly, IAX2 je ovšem zakomentován znakem „;"a pro jeho použití se musí tento znak odstranit a okomentovat příkaz vytvoření s použitím SIP protokolu.

Po dokončení konfigurace soubor uložíme a přejdeme do Asterisk CLI, kde příkazem **dialplan reload** načteme již upravený konfigurační soubor. Pro kontrolu můžeme příkazem **dialplan show** zobrazit načtený dialplan.

Podle obrázku 3 na virtuálním stroji definujeme pouze context incoming, pomocí kterého řekneme Asterisku, že má přijmout hovor příkazem Answer, dále příkazem Echo vrátit přicházející zvuk do kanálu, aby jsme jej mohli na pevném počítači zaznamenat pomocí příkazu Monitor a ukončit spojení po dokončení přehrávaní testovací zvukové stopy příkazem Hangup.

Po dokončení konfigurace uložíme soubor a v Asterisk CLI příkazem **dialplan reload** načteme již upravený konfigurační soubor, opět pro kontrolu můžeme příkazem **dialplan show** zobrazit načtený dialplan.

```
[globals]

[general]

[incoming]
exten => _X,1,Answer()
exten => _X,2,Echo()
exten => _X,3,Hangup()
```

Výpis 4: Konfigurace extensions.conf na virtuálním stroji

3.2 Provedení spojení a vyhodnocení

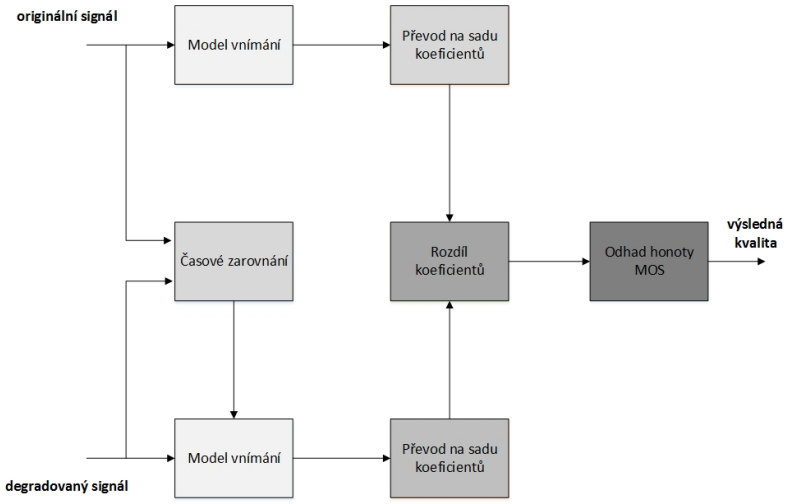
Teoretický rozbor

V této kapitole budeme přenášet testovací zvukové soubory, které zachytíme během provozu a porovnáme metodou PESQ (Perceptual Evaluation of Speech Quality), který zjistí kvalitu hovoru a vyjádří ji pomocí hodnoty MOS (Mean opinion score). Samotné zahájení hovoru se vykonává přes Asterisk CLI příkazem **originate**.

- PESQ - Porovnává originální signál X(t) s degradovaných signálem Y(t), což je originální signál po průchodu komunikačním systémem. Výsledkem ohodnocení je hodnota MOS, která udává, jakou kvalitu by měl hovor pro lidského posluchače. Tento postup lze popsat blokovým schématem, které je na obrázku 4. Tato metoda má však limit v maximální hodnotě změřené kvality, nezvládne určit kvalitu hovoru nad 4.5 MOS.[11, 16]
- MOS - Udává odhadovanou kvalitu zvuku během komunikace nebo kódování. Hodnota je udávána v rozmezí 5 (výborná kvalita) po 1 (nedostačující kvalita) a vynesena na tabulku. Tato tabulka je dlouhou řadou experimentů a je definována v ITU-T P.800. [12]

3.2.1 Provedení spojení

Když máme nakonfigurovány obě telefonní ústředny Asterisk, můžeme provést testovací hovory. Doporučuji přenášet více souborů s rozdílnou délkou několikrát po sobě, ať máme řadu výsledků, ze kterých získáme průměrnou hodnotu. Zde stejně jako v elektrotechnice budeme opakovat každý hovor desetkrát. V mém řešení mám zvukové soubory s délkou 1 minuta, 30 sekund a 20 sekund. Zvukové stopy musí byt samozřejmě v takové podobě, aby jej Asterisk dokázal přehrát (mono, 8kbit a nejlépe typ sln, který má nejlepší podporu).



Obrázek 4: Blokové schéma metody PESQ [16]

Přenášené soubory se musí nacházet v lokaci **/var/lib/asterisk/sounds/en** a jejich název se bude zadávat místo XX u příkazu Playback v dialplanu pevného počítače, který je zobrazen ve výpisu 3.

3.2.1.1 Zvukový soubor délky 1 minuta

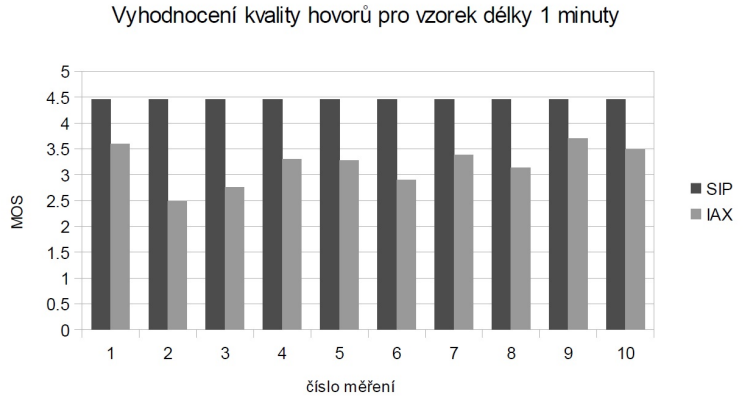
Začneme přenášením zvukového souboru délkou 1 minuty (dále jen 1m), použijeme následující postup:

1. Vložíme testovací zvukový soubor do lokace **/var/lib/asterisk/sounds/en**.
2. Otevřeme konfigurační soubor extensions.conf a přepíšeme XX na název souboru 1m, povolíme příkaz Dial pro SIP, uložíme a v Asterisk CLI provedeme načtení dialplanu.
3. V Asterisk CLI příkazem **originate local/1@incoming extension 2@outgoing** provedeme hovor a vyčkáme, než se nám ukončí. Doba hovoru závisí na délce testovacího zvuku, tento postup opakujeme desetkrát.
4. Po vytvoření deseti hovorů pomocí SIP protokolu otevřeme extensions.conf, zakomentujeme příkaz Dial pro SIP a povolíme Dial pro IAX. Soubor uložíme, načteme v Asterisku a opět provedeme komunikaci desetkrát.
5. Po provedené komunikaci si zálohujeme soubory v lokaci **/var/spool/asterisk/monitor**, které porovnáme metodou PESQ. Obsah složky vymažeme, aby se nám soubory nepomíchaly se soubory dalších hovorů.

Metoda PESQ nám vypíše hodnoty MOS pro jednotlivé spojení, v mém případě jsou výsledky uvedeny v tabulce 7 a vyneseny v grafu 5.

SIP	4.461	4.461	4.461	4.461	4.461	4.461	4.461	4.461	4.461	4.461
IAX	3.601	2.484	2.760	3.304	3.277	2.896	3.385	3.140	3.696	3.496

Tabulka 7: Výsledky MOS pro komunikaci délky 1 minuty



Obrázek 5: Graf výsledků komunikace délky 1 minuty

3.2.1.2 Zvukový soubor délky 30 sekund

Postup je stejný jako předchozí, ale nesmíme zapomenout vložit do adresáře testovací zvukový soubor, přejmenovat název souboru v dialplan, provést načtení v Asterisku a provést hovor desetkrát pro každý protokol.

Metoda PESQ nám vypíše hodnoty MOS pro jednotlivé spojení, v mém případě jsou výsledky uvedeny v tabulce 8 a vyneseny v grafu 6.

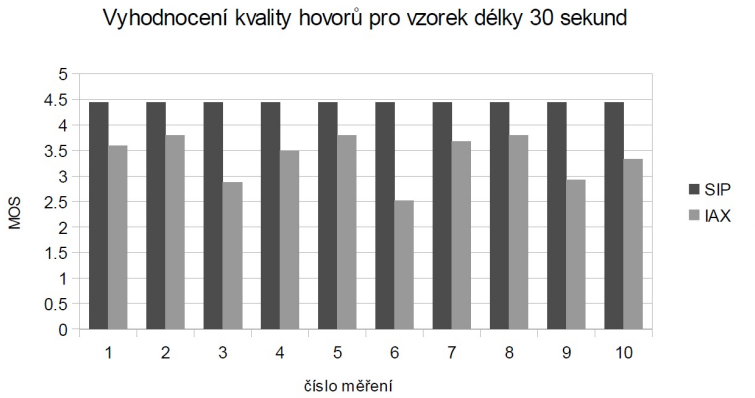
3.2.1.3 Zvukový soubor délky 20 sekund

Postup se opět neodlišuje od předchozích.

Metoda PESQ nám změří hodnoty MOS pro jednotlivé spojení, v mém případě jsou výsledky uvedeny v tabulce 9 a vyneseny v grafu 7.

SIP	4.434	4.434	4.434	4.434	4.434	4.434	4.434	4.434	4.434	4.434
IAX	3.592	3.793	2.788	3.495	3.792	2.517	3.677	3.793	2.918	3.327

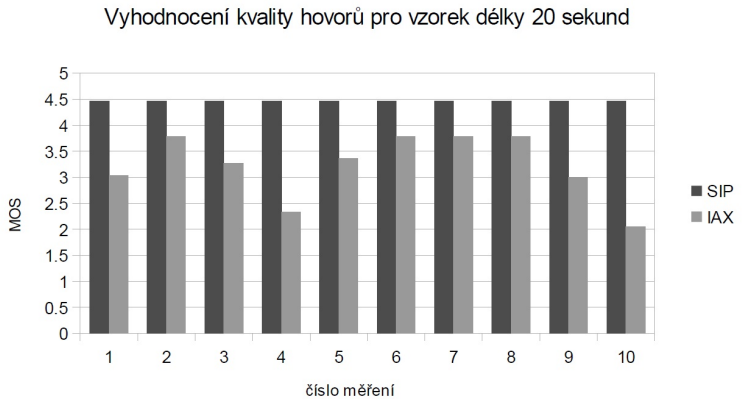
Tabulka 8: Výsledky MOS pro komunikaci délky 20 sekund



Obrázek 6: Graf výsledků komunikace délky 30 sekund

SIP	4.469	4.469	4.469	4.469	4.469	4.469	4.469	4.469	4.469	4.469
IAX	3.033	3.788	3.263	2.332	3.363	3.789	3.788	3.788	3.005	2.057

Tabulka 9: Výsledky MOS pro komunikaci délky 20 sekund



Obrázek 7: Graf výsledků komunikace délky 20 sekund

4 Závěr

V bakalářské práci byly popsány možnosti virtualizace pomocí nástrojů poskytovaných společností VMware. Byla zmíněna telefonní ústředna Asterisk, jaké má možnosti nasazení, používané protokoly a kodeky. Následuje popis protokolů SIP a IAX.

Signalizační protokol SIP je v dnešní době hojně využíván, zatímco protokol IAX patří mezi nejnovější na poli internetové telekomunikace. V práci jsou popsány způsoby vytváření komunikačního kanálu a základní vlastnosti těchto protokolů.

Výsledkem této bakalářské práce bylo vytvoření funkčního komunikačního kanálu mezi telefonní ústřednou Asterisk nainstalovanou na virtuálním stroji a ústřednou Asterisk na pevném stroji. Postup nastavení se nachází v kapitole 3.1.

Dalším bodem bylo provedení hovoru mezi ústřednami a změření jeho kvality programem PESQ (Perceptual Evaluation of Speech Quality), což se nachází v kapitole 3.2.1. Kvůli přesnosti zjištěné kvality, byly desetkrát přeneseny zvukové soubory délky 1 minuta, 30 sekund, 20 sekund pro každý protokol a z výsledných hodnot byl vytvořen průměr, který definuje kvality hovorového kanálu.

Kvality hovorů byly vyneseny do grafů, pro soubor délky 1 minuty je graf 5, hodnoty 30 sekundového souboru v grafu 6 a hodnoty přenosu délky 20 sekund se nachází v grafu 7. Při pohledu na grafy je zřetelné, že hodnoty SIP hovorů nepřesahují hodnotu MOS 4.5, což je způsobeno omezením metody PESQ. Výsledné grafy ukazují, že protokol SIP dosahuje lepší kvality přenosu než protokol IAX. Což je zapříčiněno způsobem přenosu dat IAX protokolu, který neposílá data souvisle, ale posílá je v určitém časovém intervalu, jenž je daný procesorem počítače. Vliv na výsledné hodnoty má samozřejmě i vytížení komunikační linky.

Tato bakalářská práce byla pro mne přínosem, naučil jsme se pracovat s telefonní ústřednou Asterisk, získal jsme znalosti o protokolech SIP a IAX. Znalosti získané během psaní bakalářské práce považují za přínos do budoucna, kde budou podobná řešení běžná a díky této práci budu mít potřebné znalosti pro jejich správu. Má bakalářka práce dokázala, že možnost komunikace telefonní ústředny Asterisk na virtuálním stroji s ústřednou Asterisk na pevném stroji je možná. Ovšem vyhodnocení kvality komunikace mezi ústřednami ukázalo, že dlouhodobě využívaný komunikační protokol SIP dosahuje mnohem kvalitnějšího hovoru, než protokol IAX, který má dobré vlastnosti, ale má problémy se způsobem přenosu komunikace, což zapříčinilo špatnou kvalitu hovoru. V budoucnosti by se mé řešení mohlo otestovat na dvou virtuálních strojích, jestli se i zde projeví problémy s přenosem pomocí protokolu IAX nebo je tento problém výhradně na pevném stroji. Pokud se potvrdí, že problémy s kvalitou přenosu hovoru lze odstranit použitím virtuálních strojů, získáme kvalitní komunikační protokol.

5 Reference

[1] BASTIAANSEN, Rob. Rob's guide to using VMware. 2nd ed. Netherlands: Books4Brains, 2005. ISBN 90-808-9343-9.

[2] ZIMMER, Dennis. VMware Server and VMware Player: the way forward for Virtualization. 1. Aufl. Maur: Sunny Publ, 2006. ISBN 39-522-9421-7.

[3] RYAN TROY, Matthew Helmke a Traducción Patricia SUGG. Vmware Cookbook A Real-world Guide to Effective Vmware Use: the way forward for Virtualization. 1. Aufl. Madrid: Oreilly, 2010. ISBN 14-493-1447-3.

[4] MEGGELEN, Jim Van, Leif MADSEN a Jared SMITH. Asterisk: the future of telephony. 2nd ed. Beijing: O'Reilly, 2007, 574 s. ISBN 05-965-1048-9.

[5] LEIF MADSEN, Jim Van Meggelen, Leif MADSEN a Jared SMITH. Asterisk: the definitive guide. 3rd ed. Sebastopol, CA: O'Reilly Media, Inc, 2007, 574 s. ISBN 05-965-1734-3.

[6] SINNREICH, Henry a Alan B JOHNSTON. Internet communications using SIP: delivering VoIP and multimedia services with Session Initiation Protocol. 2nd ed. Indianapolis, IN: Wiley Pub., c2006, xxix, 377 p. ISBN 978-047-1776-574.

[7] SIP: Session Initiation Protocol [online]. 2002 [cit. 2012-12-11]. Dostupné z: http://tools.ietf.org/html/rfc3261

[8] AHSON, Syed a Mohammad ILYAS. VoIP handbook: applications, technologies, reliability, and security. 2nd ed. Boca Raton: CRC Press, c2009, xv, 454 p. ISBN 14-200-7020-7.

[9] RFC 5456 - IAX: Inter-Asterisk eXchange Version 2 [online]. 2002 [cit. 2013-02-18]. Dostupné z: http://tools.ietf.org/html/rfc5456

[10] BOUCADAIR, Mohamed a Mohammad ILYAS. Inter-asterisk exchange (IAX): deployment scenarios in SIP-enabled networks. 1st ed. Chichester: John Wiley, 2009, xv, 454 p. ISBN 978-0-470-77072-6.

[11] ITU-T P.862: Perceptual Evaluation of Speech Quality (PESQ): An Objective Method for End-to-end Speech Quality Assessment of Narrow-band TelephoneNetworks and Speech Codecs, 2001.

[12] ITU-T P.800.1: Mean Opinion Score (MOS) terminology, 2003

[13] BINDER, Tomáš. Správa a konfigurace VoIP ústředny Asterisk: Management and configuration of Asterisk VoIP exchange. Brno: Vysoké učení technické, Fakulta elektrotechniky a komunikačních technologií, 2008.

[14] SCHÖN, Martin. Realizace VoIP ústředny Asterisk: Management and configuration of Asterisk VoIP exchange. Brno: Vysoké učení technické, Fakulta elektrotechniky a komunikačních technologií, 2011. 64 l.

[15] BÍLEK, Jan. Měření kvality telefonních hovorů u pobočkové ústředny Asterisk: Management and configuration of Asterisk VoIP exchange. Brno: Vysoké učení technické, Fakulta elektrotechniky a komunikačních technologií, 2011. 49, 2 l.

[16] Měření a hodnocení QoS v IP telefonii. ITPOINT [online]. [cit. 2013-04-18]. Dostupné z: http://www.itpoint.cz/voip/?i=qos-mereni-a-hodnoceni-17

[17] Voip Think- Voice over IP - Asterisk and SER - SIP IAX and H323. Voip Think - IAX IAX2 - F and M frames - Full Frames and Mini Frames [online]. [cit. 2013-04-05]. Dostupné z: http://www.en.voipforo.com/IAX/IAX-frames.php